Stationary Latent Weight Inference for Unreliable Observations from Online Test-Time Adaptation

Jae-Hong Lee [1] **Joon-Hyuk Chang** [1]

Abstract

In the rapidly evolving field of online test-time adaptation (OTTA), effectively managing distribution shifts is a pivotal concern. State-of-the-art OTTA methodologies often face limitations such as an inadequate target domain information integration, leading to significant issues like catastrophic forgetting and a lack of adaptability in dynamically changing environments. In this paper, we introduce a stationary latent weight inference (SLWI) framework, a novel approach to overcome these challenges. The proposed SLWI uniquely incorporates Bayesian filtering to continually track and update the target model weights along with the source model weight in online settings, thereby ensuring that the adapted model remains responsive to ongoing changes in the target domain. The proposed framework has the peculiar property to identify and backtrack nonlinear weights that exhibit local non-stationarity, thereby mitigating error propagation, a common pitfall of previous approaches. By integrating and refining information from both source and target domains, SLWI presents a robust solution to the persistent issue of domain adaptation in OTTA, significantly improving existing methodologies. The efficacy of SLWI is demonstrated through various experimental setups, showcasing its superior performance in diverse distribution shift scenarios.

1. Introduction

Deep neural networks (DNNs) have achieved remarkable success across various applications, particularly in computer vision and speech recognition (Krizhevsky et al., 2012; Simonyan & Zisserman, 2014; He et al., 2016; Hinton et al., 2012; Graves et al., 2013). The success of DNNs predominantly depend on the assumption that the training and testing datasets encompass independent and identically distributed (i.i.d) samples under the same distribution (Goodfellow et al., 2016; Murphy, 2023). However, this assumption can be easily invalidated in real-world scenarios because of minor changes such as varying weather conditions or noise introduced by aging sensors (Hendrycks & Dietterich, 2019b; Koh et al., 2021). Moreover, data collected in real-world environments often exhibit biases towards specific environments or labels owing to limitations in data collection methodologies. Therefore, the performance of DNNs degrades significantly when there are distribution shifts (Quinonero-Candela et al., 2008; Sun et al., 2017), and addressing domain adaptation under varying distribution shifts becomes a crucial issue.

Online test-time adaptation (OTTA) has emerged as a potent solution to domain adaptation in distribution shifts. It involves concurrent training and testing on the streaming data from the target domain where distribution shifts occur. The OTTA framework performs unsupervised source-free domain adaptation. As OTTA operates in real-time, relying on human annotation for training is impractical, necessitating an unsupervised learning approach. Furthermore, access to source-domain data is often restricted, and only models pre-trained on source data are available. Owing to these limitations, OTTA methods employ unsupervised objective functions for the source model. In general, test entropy minimization (TENT) (Wang et al., 2020) has demonstrated the effectiveness of entropy minimization in single domains. However, recent studies have shown that entropy minimization often fails under more diverse domain shifts, such as in multiple domains (Boudiaf et al., 2022; Zhang et al., 2022; Chen et al., 2022). In distribution shifts, an overlap of information with particular domains can occur, leading to model overfitting for specific information and subsequently forgetting previous knowledge. This results in catastrophic forgetting of the target models that are adapted to the target data.

To mitigate catastrophic forgetting, recent OTTA methods

[1] Department of Electronic Engineering, Hanyang University, Seoul, Republic of Korea. Correspondence to: Jae-Hong Lee <ljh93ljh@hanyang.ac.kr>, Joon-Hyuk Chang <jchang@hanyang.ac.kr>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

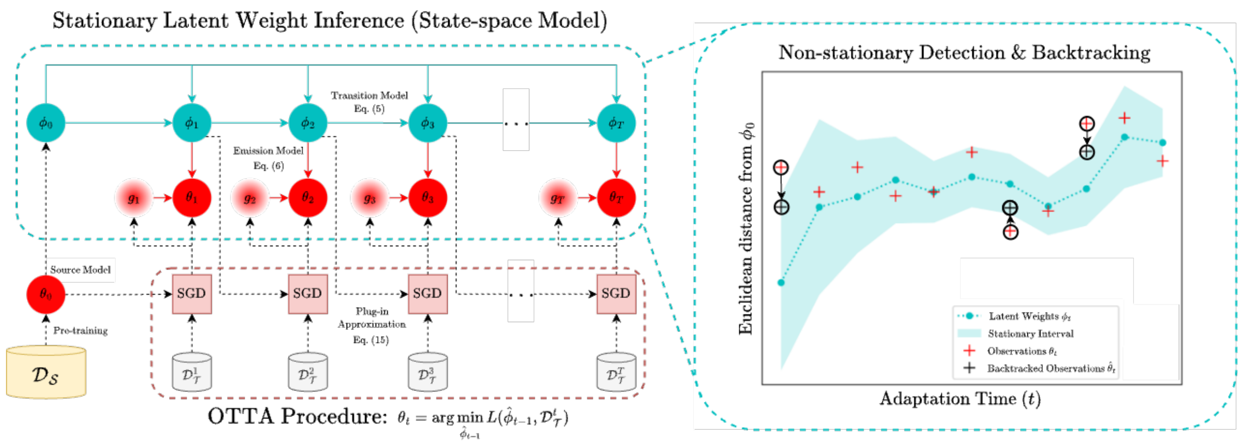


Figure 1. Illustration of the proposed stationary latent weight inference (SLWI) framework. The OTTA procedure starts with a target and latent models initialized using a source model pre-trained on source data $\mathcal{D}_S$. During adaptation, target weights θ_t are obtained via stochastic gradient descendent(SGD)-based optimization on the target data $\mathcal{D}_T^t$ for each time step t. These weights serve as observations for SLWI, which updates the latent weight ϕ_t to accumulate information from both the source weight $\theta_0 = \phi_0$ and target weight using our Bayesian filtering (as depicted in the state-space model). Nonlinear weights, which deviate from the posterior predictive distribution of observations, are identified and backtracked for local stationary processes (illustrated in the right graph). The proposed SLWI concludes by transferring the refined latent weight information to the OTTA procedure via plug-in approximation represented in Eq. (15). This SLWI intercepts the target weight from the general OTTA framework (Algorithm 1) before the weight is passed to the next time step's SGD, refines it, and then transfers it back to the SGD. The SLWI framework presents in Algorithm 2.

adopted a dual approach: sample filtering and leveraging information from a source model (Wang et al., 2022; Niu et al., 2022; 2023; Marsden et al., 2023). Sample filtering involves identifying low-confidence samples by measuring the entropy of model outputs. Despite these measures, the accuracy of the source model can be compromised, allowing the recovery of the target model using source model information during online adaptation. These methods prevent excessive overfitting of specific target domains by continuously transmitting weight-related information related to the weights (i.e., parameter of DNNs) extracted from the source model to the target model. However, they often neglect information learned from the target domain and are heavily dependent on the source model.

In this paper, we propose the stationary latent weight inference (SLWI) framework, which is an innovative approach that enhances the domain adaptation capabilities by continually accumulating information from both the source and target domains (Figure 1). This accumulation is facilitated by a latent model that integrates and refines the dual-domain information. The core of the proposed framework lies in designing the dynamics of Bayesian filtering, which updates the latent weight by tracking the target weights (i.e., observations) along with the source weights in real time. This approach ensures that the model is responsive and adaptable to the ongoing changes in the target domain. A key challenge SLWI addresses is the potential error propagation resulting from accumulating unreliable target weights obtained from unsupervised objective functions. To counter this, we incorporate a novel mechanism for identifying and backtracking nonlinear weights that exhibit local non-stationarity by leveraging the posterior predictive distribution of the target weights, which is naturally derived from our framework. The proposed SLWI framework effectively identifies and corrects instances in which accumulated weights may lead to erroneous behaviors in the model, thereby enhancing the reliability and stability of the OTTA procedure.

The main contributions of this study can be summarized as follow:

- We provide a framework that combines target-domain information with source-domain information via latent models and Bayesian filtering with probabilistic interpretation.
- The proposed framework effectively prevents error propagation caused by the nonlinearity of the unreliable target weights obtained during the OTTA procedure.
- The SLWI framework seamlessly integrates with existing OTTA methods and requires minimal additional computational resources.
- The proposed framework exclusively utilizes model weights, negating the need to store data from the target domain, resulting in inherently safeguarding privacy concerns.

We evaluate the proposed framework across a multitude of distribution-shift scenarios and datasets, which show significant performance improvements compared with current state-of-the-art methods.

2. Information Retention in the Online Test-Time Adaptation Procedure

2.1. OTTA Procedure

Let us denote the input data as $\mathbf{x} \in \mathbb{R}^d$, where d denotes the dimension. The model output is denoted by $\mathbf{z} \in \mathbb{R}^K$ for K classes and the label is represented as $y \in \mathbb{R}$. We assume a well-pre-trained source model $f(\cdot, \phi_0) : \mathbb{R}^d \rightarrow \mathbb{R}^K$ with initial weights $\phi_0 = \theta_0$, trained on source data $(\mathbf{x}_n, y_n) \sim \mathcal{D}_{\mathcal{S}}$. This model is updated up to time step t on the streaming target data $(\mathbf{x}^t_n) \sim \mathcal{D}^t_{\mathcal{T}}$, resulting in the target model $f(\cdot, \theta_t)$.

Because OTTA methods do not allow ground-truth labels y^t_n for adaptation, an unsupervised objective function $\ell : \mathbb{R}^K \rightarrow \mathbb{R}$ is adopted (Wang et al., 2020; Liang et al., 2020). This function utilizes output $\mathbf{z}^t_n$ instead of predicted label $\hat{y}^t_n$ and is based on entropy. The total objective function is formulated as follows:

$$L(\theta_{t-1}, \mathcal{D}^t_{\mathcal{T}}) = \frac{1}{|\mathcal{D}^t_{\mathcal{T}}|} \sum_{\mathbf{x^t_n} \in \mathcal{D}^t_{\mathcal{T}}} \ell \left(\log p(\mathbf{z}^t_n | \mathbf{x}^t_n, \theta_{t-1}), M(\mathbf{z}^t_n) \right), \tag{1}$$

where $M(\mathbf{z}^t_n)$ is a binary mask that filter the low-confidence samples based on entropy-driven confidence scores (Niu et al., 2022; 2023; Marsden et al., 2023). Further refinement of these scores involves data augmentation (Wang et al., 2022; Yuan et al., 2023; Marsden et al., 2023). The optimization to minimize the total loss of the target weight at each time step is given by:

$$\theta_t = \arg\min_{\theta_{t-1}} L(\theta_{t-1}, \mathcal{D}^t_{\mathcal{T}}) + \mathcal{R}(\phi_0, \theta_{t-1}), \tag{2}$$

where $\mathcal{R}(\phi_0, \theta_{t-1})$ is a regularization term that ensures that the target model weight θ_{t-1} does not significantly diverge from the source model weight ϕ_0. This optimization is typically performed by an SGD-based optimizer (Ruder, 2016). The model predictions $\hat{y}_t$ obtained from the target model are immediately evaluated against the ground-truth labels. Moreover, the model weights obtained from this process serve as the initial weights for subsequent optimization, thus facilitating continuous online domain adaptation.

2.2. Source Model Information Retention

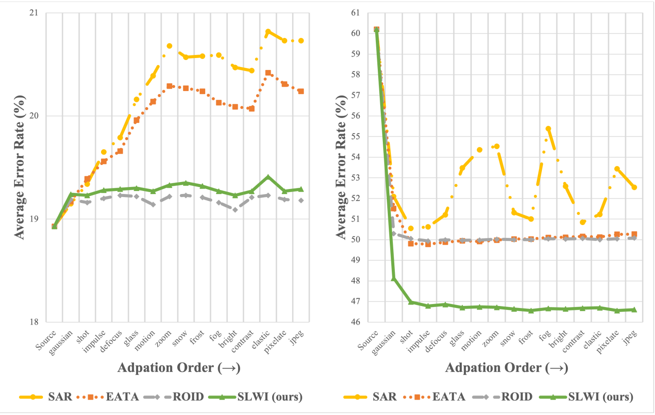
The introduction of the regularization term $\mathcal{R}(\phi_0, \theta_{t-1})$ play a critical role in enhancing the reliability of the adaptation process. This term is especially pivotal in addressing the limitations of filtering methods integrated into the OTTA objective function. These limitations arise from the reliance on scores extracted from the source model, which may significantly underperform in the target domain, leading to error propagation and catastrophic forgetting. The challenge of catastrophic forgetting, in which the target model gradually loses crucial the source model information, is a significant concern in OTTA. This is primarily due to the reliance on scores from a source model that performs poorly in the target domain, leading to error propagation. Strategies that continuously transfer information from the source model to the target model have been developed to address this issue.

Several studies (Wang et al., 2022; Yuan et al., 2023; Niu et al., 2022; 2023; Marsden et al., 2023) have proposed ensuring retraining source model information in the target model. EATA (Niu et al., 2022) employs the Fisher information matrix (Kirkpatrick et al., 2017), which is calculated using the source model and data to determine the importance of weights. This approach ensures that parts of the target weight θ_t do not excessively deviate from the source model ϕ_0. SAR (Niu et al., 2023) used a different approach in which the target weight is reset to the source weight if the exponential moving average of the total loss falls below a certain threshold. This strategy serves as a reset mechanism. ROID (Marsden et al., 2023) adopts a continuous constraint approach by consistently averaging the source and target weights using a weight ensemble (Guo et al., 2023). These strategies play a crucial role in stabilizing the OTTA procedure. This prevents the target model from overfitting to a specific target domain and loses valuable source information. This stability is vital for ensuring the efficacy and reliability of the OTTA process, particularly in dynamic environments where the target domain is continuously evolving.

2.3. Limitation of Source Model Dependency

A critical limitation inherent to state-of-the-art methods is their continuous source-model dependency, mainly due to regularization strategies. This dependence is evident in Eq. (1), where the probability calculation considers only the weight from the previous time step θ_{t-1}. Consequently, these strategies neither incorporate the newly learned target weights $\theta_{1:t-2}$ from different domains nor effectively leverage ongoing information from the target domain.

This limitation is illustrated in Figure 2, which shows the average error rates of state-of-the-art OTTA methods across various domains according to the adaptation order. The average error rate is measured post-adaptation using the complete dataset of the source or target domain. As shown in Figure 2 (a), most existing methods tend to converge while preserving the source information effectively, with error rates between 1% and 2%. This trend indicates an acceptable retention of source information retention.



(a) Source domain

(b) Target domain

Figure 2. Illustration of the information retention ability of state-of-the-art OTTA methods on ImageNet-C in covariate shifts for ViT. Average error rates (%) are measured on the entire dataset of the source or target domain after the adaptation for each domain was completed according to the adaptation order.

Algorithm 1 General OTTA Framework

```
Input: Input data stream $\{\mathcal{D}_T^1, \dots, \mathcal{D}_T^T\}$, Source model $f(.;\theta_0)$
Output: $\theta_T$
Initialization $\phi_0 \leftarrow \theta_0$
for $t = 1$ to $T$ do
    OTTA procedure:
    $\theta_t \leftarrow \arg\min_{\theta_{t-1}} L(\theta_{t-1}, \mathcal{D}_T^t) + \mathcal{R}(\phi_0, \theta_{t-1})$
end for
```

However, Figure 2 (b) shows a starkly different trend in the target domain. After the first domain, represented by Gaussian noise, the performance of existing methods in the subsequent domains shows minimal improvement or even divergence. This trend is concerning because it suggests a lack of adaptability to the evolving target domain. These results highlight a fundamental issue in OTTA methods: their dependency on source model information and their inability to continuously integrate target domain information. In other words, existing methods fail to evolve and enhance their capabilities by accumulating target domain information. As the number of target domains increases, this limitation becomes increasingly pronounced, hindering the potential for significant performance improvements in evolving environments.

3. Stationary Latent Weight Inference

3.1. Overview

The primary goal of this paper is to enhance the forward transferability to the target domain while mitigating catastrophic forgetting of past information, including the source domain. To achieve this, we introduce a latent weight, ϕ_t, which is updated using the source weight $\phi_0 = \theta_0$ and the target weight θ_t, aiming to reduce the dependency on the source model. The target weight is obtained in an unsupervised manner, which inherently includes noise. Consequently, updating the latent weight with the target weight poses a risk of error propagation. To counter this, we utilize our Bayesian filtering method, which possesses noise reduction capabilities, to update the latent weight (Section 3.2). Our Bayesian filtering maintains a local stationary process through strategies for detecting and backtracking rapidly changing nonlinear weights (Section 3.3). These strategies ensure that the observations (i.e., target weights) conform to the linear Gaussian model assumed by Bayesian filtering.

Algorithm 2 Stationary Latent Weight Inference Framework

```
Input: Input data stream $\{\mathcal{D}_T^1, \dots, \mathcal{D}_T^T\}$, Source model $f(.;\theta_0)$, Bayesian filtering parameters $\Omega = (a, b, q)$, Hyperparameters $\omega = (\zeta, \alpha)$, Learning rate $\alpha_0$
Output: $\hat{\phi}_T$
Initialization $\phi_0 \leftarrow \theta_0$, $\hat{\phi}_0 \leftarrow \theta_0$, $\mu_{0|0} \leftarrow \theta_0$, $\sigma^2_{0|0} \leftarrow 0$, $\log p_{g,t-1}(\theta_{0:0}) \leftarrow 0$, $\Delta^2_{0:0} \leftarrow 0$
for $t = 1$ to $T$ do
    OTTA procedure:
    $\theta_t \leftarrow \arg\min_{\hat{\phi}_{t-1}} L(\hat{\phi}_{t-1}, \mathcal{D}_T^t)$
    $g_t \leftarrow \alpha_0 \nabla_{\hat{\phi}_{t-1}} L(\hat{\phi}_{t-1}, \mathcal{D}_T^t)$
    Bayesian Filtering & Backtracking:
    $\mu_{t|t}, \mu_{t|t-1}, \sigma^2_{t|t}, \log p_{g,t-1}(\theta_{0:t}), \Delta^2_{0:t} \leftarrow$ FILTER($t$,
                                      $\theta_t, g_t, \phi_0,$
                                      $\mu_{t-1|t-1}, \sigma^2_{t-1|t-1},$
                                      $\log p_{g,t-1}(\theta_{0:t-1}), \Delta_{0:t-1};$
                                      $\Omega, \omega$)
    ▷ Algorithm 3 for FILTER.
    Plug-in Approximation:
    $\_, b, \_ \leftarrow \Omega$
    $\hat{\phi}_t \leftarrow \mu_{t|t-1} + b\,(\hat{\theta}_t - \mu_{t|t-1})$        ▷ Eq. (15)
end for
```

Through our Bayesian filtering, we compute the posterior distribution of ϕ_t and transfer this information to the OTTA procedure. The predictive distribution for $\mathbf{z}_t$ is calculated as follows:

$$
\begin{aligned}
&p(\mathbf{z}_t|\mathbf{x}_t, \theta_{1:t-1}, g_{1:t-1}, \phi_0) \\
&\quad = \int p(\mathbf{z}_t|\mathbf{x}_t, \phi_{t-1}) p(\phi_{t-1}|\theta_{1:t-1}, g_{1:t-1}, \phi_0) d\phi_{t-1},
\end{aligned} \tag{3}
$$

where g_t is time-varied auxiliary variable. This result omits $\mathcal{R}(\phi_0, \theta_{t-1})$ from Eq. (2) and substitutes the model output probability $p(\mathbf{z}_t|\mathbf{x}_t, \theta_{t-1})$ with $p(\mathbf{z}_t|\mathbf{x}_t, \theta_{1:t-1}, g_{1:t-1}, \phi_0)$. This substitution is part of our strategy to integrate information from both the source and target domains more effectively.

Given the significant computational cost associated with the calculation $p(\mathbf{z}_t|\mathbf{x}_t, \phi_{t-1})$, direct computation in an OTTA scenario is impractical. Therefore, we employ a widely applicable plug-in approximation. This approach uses a point-estimated weight, $\hat{\phi}_{t-1} = \arg\max_{\phi_{t-1}} p(\phi_{t-1}|\theta_{1:t-1}, g_{1:t-1}, \phi_0)$, and a delta function, $\delta(.)$, which leads to $p(\phi_{t-1}|\theta_{1:t-1}, g_{1:t-1}, \phi_0) \approx \delta(\phi_{t-1} - \hat{\phi}_{t-1})$. Consequently, the posterior predictive distribution is approximated as follows:

$$\begin{aligned} p(\mathbf{z}_t|\mathbf{x}_t, \theta_{1:t-1}, g_{1:t-1}, \phi_0) \\ \approx \int p(\mathbf{z}_t|\mathbf{x}_t, \phi_{t-1})\delta(\phi_{t-1} - \hat{\phi}_{t-1})d\phi_{t-1} \\ = p(\mathbf{z}_t|\mathbf{x}_t, \hat{\phi}_{t-1}). \end{aligned} \tag{4}$$

Using the posterior derived from Eq. (4), we can inject information from both domains into the OTTA procedure (Section 3.4). As a result, the proposed framework is executed step-by-step as shown in Algorithm 2. This process is regarded as an extension of the general framework adopted in existing OTTA methods (Algorithm 1), where the regularization term for the source model is replaced with the posterior distribution of the latent weight derived from our Bayesian filtering using the plug-in approximation for the regularization of the next time step's OTTA procedure. More details of the step-by-step implementation, theoretical background, and derivations are provided in Appendix A.

3.2. Bayesian Filtering for Unreliable Weights

The OTTA framework operates on an unsupervised learning paradigm utilizing a source model that often exhibits suboptimal performance in the target domain. This results in observations, specifically, the target weights, which are fundamentally unreliable and noisy. We adopt Bayesian filtering as our latent inference framework to address the intrinsic noise associated with observations, resulting in the suppression of observation noise. The framework is structured around linear Gaussian models that imply potential error through variance, approximated by the Gaussian assumed density approximation (Särkkä & Svensson, 2023).

Bayesian filtering comprises a transition model and an emission model, both of which are recursively applied to infer the posterior distribution. The transition model predicts the current latent weight based on the previous latent weight, while the emission model tracks the observations using the predicted latent weight. First, we parameterize the transition model as follows:

$$p(\phi_t|\phi_{t-1}, \phi_0) = \mathcal{N}(\phi_t|a\phi_{t-1} + (1-a)\phi_0, q), \tag{5}$$

where $a, q \in \mathbb{R}$. The transition model variance is $0 \leq q < 1$, which determines the degree of change in the latent weight and $0 \leq a < 1$ adjusts the extent of recovery of the latent weight. Unlike typical emission models, our approach uses recovery strategies to prevent latent weight from being corrupted by unreliable observations. Our parameterized emission model is formulated as follows:

$$p(\theta_t|\phi_t, g_t) = \mathcal{N}(\theta_t|\phi_t - (1-c_t)g_t, 1-q), \tag{6}$$

where $g_t = \alpha_0 \nabla_{\hat{\phi}_{t-1}} L(\hat{\phi}_{t-1}, \mathcal{D}_{\mathcal{T}}^t)$ with respect to the learning rate $\alpha_0 \in \mathbb{R}$. The initial condition is $c_t = 1$, and $p(\theta_t|\phi_t, g_t) = p(\theta_t|\phi_t)$. The variance is set to $1-q$, inversely related to the transition model. The emission model tracking unreliable observations exhibits a larger variance than the transition model. The degree of backtracking $0 \leq c_t \leq 1 \in \mathbb{R}$, which reflects the nonlinearity of the observations, allows the mean of the emission model to track extremely varying observations, justifying a stationary variance.

Given the linear Gaussian models and the previous posterior $p(\phi_{t-1}|\theta_{1:t-1}, g_{1:t-1}, \phi_0) = \mathcal{N}(\phi_{t-1}|\mu_{t-1|t-1}, \sigma^2_{t-1|t-1})$, we calculate the posterior in the current time step by Bayesian inference as follows:

$$p(\phi_t|\theta_{1:t}, g_{1:t}, \phi_0) = \mathcal{N}(\phi_t|\mu_{t|t}, \sigma^2_{t|t}), \tag{7}$$

$$\kappa_t = \sigma^2_{t|t-1}(a^2\sigma^2_{t-1|t-1} + 1)^{-1}, \tag{8}$$

$$\mu_{t|t} = \mu_{t|t-1} + \kappa_t\,(\hat{\theta}_t - \mu_{t|t-1}), \tag{9}$$

$$\sigma^2_{t|t} = \sigma^2_{t|t-1} - \kappa_t\sigma^2_{t|t-1}, \tag{10}$$

where $\mu_{t|t-1} = a\mu_{t-1|t-1} + (1-a)\phi_0$, $\sigma^2_{t|t-1} = a^2\sigma^2_{t-1|t-1} + q$ and $\hat{\theta}_t = \theta_{t-1} - c_t g_t$. The initial conditions are $\mu_{0:0} = 0$ and $\sigma^2_{0:0} = 0$. Eqs. (8, 9, 10) are the same as those in the update step in the Kalman filter for the discrete-time linear Gaussian state-space model. The Kalman gain κ_t controls the degree to which the error between backtracked observation $\hat{\theta}_t$ and predicted mean $\mu_{t|t-1}$ is updated. It is important to emphasize that the dynamics for adjusting the observations based on the value of c_t are derived from our emission model. These dynamics adjust the current observation closer to the past observations as c_t decreases. Consequently, these dynamics can be utilized to backtrack nonlinear weights.

3.3. Non-Stationarity Detection and Backtracking

Target weights prone to catastrophic forgetting tend to exhibit significant local changes (Niu et al., 2023; Gong et al., 2023). Furthermore, observations that undergo rapid changes destabilize the system by violating the assumption of stationary variance in linear Gaussian models. The nonlinear weights can be adjusted using c_t. We begin by defining a score for detecting local non-stationarity and measuring temporal changes in the weight space where the initial condition is $c_t = 1$. Given by Eqs. (5, 6) and the previous posterior, the posterior predictive probability of the current weight is marginalized as follows:

$$\begin{aligned} p_{g,t-1}(\theta_t|\theta_{0:t-1}) &:= p(\theta_t|\theta_{0:t-1}, g_{1:t-1}) \\ &= \int p(\theta_t|\phi_t)p(\phi_t|\theta_{0:t-1}, g_{1:t-1})d\phi_t \\ &= \mathcal{N}(\theta_t|\mu_{t|t-1}, a^2\sigma^2_{t-1|t-1} + 1). \end{aligned} \tag{11}$$

The probability of the entire set of weights up to the current time step can be calculated using the posterior predictive as follows:

$$\log p_{g,t-1}(\theta_{0:t}) = \log p(\theta_1|\theta_0) + \sum_{\tau=2}^{t} \log p_{g,t-1}(\theta_\tau|\theta_{0:\tau-1}), \tag{12}$$

where $p_{g,t-1}(\theta_{0:t}) = p(\theta_{0:t}|g_{1:t-1})$ following by definition of Eq. (11). We now define the stationary score for $t > 1$ as the difference between the posterior predictive probabilities:

$$S(t, \Delta_t, \Delta^2_{0:t-1}) = \frac{\Delta_t}{\sqrt{\frac{1}{t-1}\Delta^2_{0:t-1}}}, \tag{13}$$

where $\Delta_t = \log p_{g,t-1}(\theta_t|\theta_{0:t-1}) - \frac{1}{t}\log p_{g,t-1}(\theta_{0:t-1})$ and $\Delta^2_{0:t-1} = \sum_{\tau=1}^{t-1}\Delta^2_\tau$ with the initial condition $\log p_{g,t-1}(\theta_{0:0}) = 0, \Delta^2_{0:0} = 0$. Our stationary score compares how the current weight differs from that of the previous time steps and how consistently the weight changes over time. The numerator represents how different θ_t are from the average trend and the denominator measures the average variability of the weights over time. This score is normalized to allow comparisons across different time steps such as batch normalization (Ioffe & Szegedy, 2015). Therefore, we use this score to compare the relative change in weights across different models or time steps. Based on this score, c_t is determined as follows:

$$c_t = \begin{cases} 1, & \text{if } |S(t, \Delta_t, \Delta^2_{0:t-1})| < \zeta \\ 1/\alpha, & \text{otherwise} \end{cases}, \tag{14}$$

where $\zeta \in \mathbb{R}$ is the stationary level and $\alpha \in \mathbb{R}$ is the degree of backtracking, which has a value greater than 1. Here, ζ assumes a form similar to the confidence level of a normal distribution. Therefore, we typically set $\zeta = 2$ to establish a confidence interval 95%. This approach detects nonlinear weights by identifying situations in which the current weight significantly deviates from the average trend, as indicated by the interval. Then, we reduces the change in the current weight by using c_t in Eq. (9), thereby guiding the observations towards stationarity. Consequently, our framework naturally derives a metric for backtracking, effectively mitigating the instability caused by nonlinear weights.

3.4. Transferring Latent Information

To transfer inferred latent weight information to the OTTA procedure, we introduce a plug-in approximation. As indicated in Eq. (10), for the approximation to be valid, the latent posterior variance $\sigma^2_{t|t}$ must be sufficiently small, similar to the variance of $\delta(.)$. Achieving this requires setting κ_t close to unity, which may not always be feasible depending on the values of q and $\sigma^2_{0:0}$. To maintain a smaller posterior variance, we introduce the transfer model $p(\theta_t|\phi_t) = \mathcal{N}(\theta_t|\phi_t, r_t)$. This model resembles the emission model, but is distinctive in that it uses $r_t \in \mathbb{R}$ and does not update the latent weight. The Kalman gain for the transfer model is derived in manner same to Eq. (8), which results in $\sigma^2_{t|t-1}(\sigma^2_{t|t-1} + r_t)^{-1}$. By setting r_t to be proportional to $\sigma^2_{t|t-1}$, we can derive the time-independent Kalman gain b. Based on the derivation of Eq. (9), the point-estimated weight is calculated as follows:

$$\hat{\phi}_t = \mu_{t|t-1} + b(\hat{\theta}_t - \mu_{t|t-1}). \tag{15}$$

This dynamic allows the direct setting of the Kalman gain close to unity by adjusting the value of $b \in \mathbb{R}$. Consequently, it enables the consistent enforcement of a sufficiently slight variance in the posterior. This feature is crucial for ensuring the effective and efficient transfer of latent weight information into the OTTA procedure and maintaining the integrity and stability of the adapted model.

4. Experiments

In all experiments, we strictly adhered to established benchmarks (Marsden & Döbler, 2022). The average and standard deviation of error rates for random seeds 0-4 were used as the evaluation metrics. More details of the experimental setup are provided in the Appendix B.

4.1. Experimental Setup

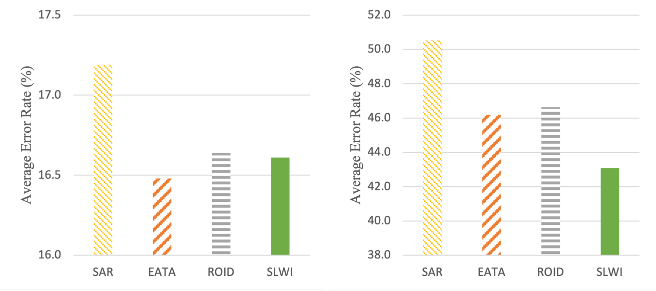
Datasets We conducted experiments on two standard datasets, ImageNet-C (Hendrycks & Dietterich, 2019a) and D109 (Marsden et al., 2023), which represent corruption and natural distribution shifts that occur in the wild-world (Niu et al., 2023). ImageNet (Deng et al., 2009) contains 1,281,167 training images and 50,000 test images. ImageNet-C is a derivative of ImageNet and was subjected to 15 types of corruption, each with five severity levels. This dataset is a standard TTA benchmark for assessing model robustness against image corruption. We selected the most severe level of corruption, that is, Level 5. D109 encompasses natural shifts across five domains and is based

Table 1. Average error rates (%) and standard deviations in **the covariate shifts scenario on ImageNet-C**. Red fonts indicate performance degradation with respect to Source.

Method	Adaptation Order (→)															Avg.
	NOISE			BLUR				WEATHER				DIGITAL				
	gaussian	shot	impulse	defocus	glass	motion	zoom	snow	frost	fog	bright	contrast	elastic	pixelate	jpeg	
Source	43.9	43.3	43.4	69.7	78.3	59.6	69.1	40.1	44.3	36.3	26.5	50.6	67.6	60.6	43.4	51.8
TENT	43.8	42.9	43.1	70.1	77.9	59.4	69.3	42.2	48.7	45.9	28.9	50.4	68.0	63.3	45.3	53.3±0.22
LAME	45.4	43.7	44.5	72.1	90.7	60.6	89.3	91.5	96.5	99.7	26.7	95.9	96.1	63.3	44.7	70.7±0.14
RoTTA	43.9	43.3	43.3	69.7	77.8	59.4	68.7	39.8	42.5	35.8	26.2	49.7	66.6	60.1	43.3	51.3±0.01
SAR	44.2	43.8	43.7	69.7	77.5	57.1	66.8	41.2	41.4	41.9	26.3	48.2	64.3	57.1	41.9	51.0±0.12
EATA	44.0	43.1	43.4	69.6	74.8	57.8	66.8	40.6	48.3	51.1	26.7	50.1	62.8	56.6	41.2	51.8±0.08
ROID	42.8	40.5	40.1	64.1	64.7	50.6	57.6	37.0	36.7	**31.8**	24.6	40.0	57.0	48.0	37.1	44.8±0.04
SLWI	**42.4**	**39.8**	**39.3**	**62.2**	**59.7**	**48.1**	**53.1**	**36.3**	**34.4**	32.0	**23.1**	**37.7**	**51.1**	**44.0**	**34.6**	**42.5±0.03**

Table 2. Average error rates (%) and standard deviations in **the covariate shifts scenario on D109**. Red fonts indicate performance degradation with respect to Source.

Method	Adaptation Order (→)					Avg.
	clipart	infograph	painting	real	sketch	
Source	48.7	72.9	41.2	20.5	56.7	48.0
TENT	49.1	77.5	51.4	31.2	79.4	57.7±0.08
LAME	98.7	99.6	96.4	51.3	99.1	89.0±0.14
RoTTA	48.6	72.6	40.7	20.0	53.9	47.2±0.03
SAR	48.3	74.4	42.9	20.3	56.5	48.5±0.10
EATA	48.1	71.8	39.5	19.6	55.1	46.8±0.03
ROID	44.0	68.9	37.7	19.3	51.2	44.2±0.06
SLWI	**41.8**	**65.7**	**35.9**	**18.4**	**46.9**	**41.8±0.05**



(a) Source domain **(b) Target domain**

Figure 3. Aaverage error rates for the source and target datasets after adaptation to all domains of ImageNet-C.

on DomainNet (Peng et al., 2019), which includes 109 classes that overlap with ImageNet.

Scenarios The domain adaptation problem in OTTA is categorized into various scenarios based on the type of distribution shifts (Zhou & Levine, 2021; Press et al., 2024; Döbler et al., 2023). The most common scenario involves *covariate shifts* in time-correlated domains (Wang et al., 2022; Yuan et al., 2023). In this scenario, the domains in each dataset were sequenced over time and the input data was streamed per domain. In addition to covariate shifts, OTTA methods consider *label shifts* (Boudiaf et al., 2022; Gong et al., 2022; Niu et al., 2023; Zhou et al., 2023). This scenario simulates the appearance of input data belonging to the same class in a time-correlated manner across the ordered domains. The simulation is based on the adjustment of the parameter γ in the Dirichlet distribution, where values closer to zero concentrate the local label distribution on specific classes.

Implementation Details In the experiments, we adopted data2vec (D2V) (Baevski et al., 2022), a self-supervised version of VisionTransformer (ViT) (Dosovitskiy et al., 2020), which is commonly used in previous studies (Niu et al., 2023; Marsden et al., 2023), as our default backbone. We also considered SwinTransformer (Swin) (Liu et al., 2021) as an additional model architecture. All models are the base size versions. The source models for each architecture were publicly available finetuned models on ImageNet for reproducibility. Following previous studies (Li et al., 2018; Mancini et al., 2018; Niu et al., 2022; 2023; Marsden et al., 2023), we designated trainable weights according to the type of normalization layers used in each model. We compared the proposed SLWI framework with the following: TENT, LAME (Boudiaf et al., 2022), RoTTA (Yuan et al., 2023), SAR, EATA, and ROID. We referred to the official implementations and hyperparameters reported in the original reports for all the comparison methods, adhering to established benchmarks (Marsden & Döbler, 2022). We set the batch size to 64, the learning rate to 0.000014, and trained the models using an SGD optimizer. Unless specifically mentioned, the ROID objective function was primarily adopted for our framework. The SLWI parameters, (a, q), were set to $(0.99, 0.001)$, and the strict Kalman gain b for the transfer model was set to the same value as a, that is 0.99. The degree of backtracking α was set to 1.4.

4.2. Comparison with Existing OTTA Methods

Table 1 presents the performances of various OTTA methods under covariate shifts scenarios on ImageNet-C. SLWI notably outperformed the existing OTTA methods in all corruption domains except for the *fog* domain. Compared with the best-performing method, i.e., ROID, SLWI exhibited a substantial performance improvement of 2.3%. Figure 3 further highlights that SLWI achieved the lowest average

Table 3. Average error rates (%) and standard deviations in **the label shifts scenario on ImageNet-C**. Red fonts indicate performance degradation with respect to Source.

		Adaptation Order (→)															
γ	Method	NOISE			BLUR				WEATHER				DIGITAL				Avg.
		gaussian	shot	impulse	defocus	glass	motion	zoom	snow	frost	fog	bright	contrast	elastic	pixelate	jpeg	
–	Source	43.9	43.3	43.4	69.7	78.3	59.6	69.1	40.1	44.3	36.3	26.5	50.6	67.6	60.6	43.4	51.8
0.0	TENT	44.1	43.7	44.0	71.1	79.2	61.6	69.8	43.2	53.1	55.9	30.8	48.7	69.4	69.1	58.9	56.2±0.98
	LAME	30.5	29.8	30.2	49.4	62.4	39.9	50.3	31.3	34.3	31.4	22.7	39.9	55.9	41.4	34.5	38.9±0.07
	RoTTA	43.8	42.0	42.0	69.8	74.5	59.3	67.4	40.2	39.5	40.1	29.0	74.1	72.3	72.8	51.4	54.5±0.03
	SAR	44.2	41.8	41.0	67.8	72.0	54.8	63.6	39.2	39.1	38.3	25.6	43.7	63.6	51.2	38.0	48.3±0.15
	EATA	44.5	43.0	43.0	65.6	71.5	53.6	62.3	39.8	41.9	40.4	24.2	43.1	58.9	52.8	39.0	48.2±0.34
	ROID	12.2	11.8	11.5	33.6	35.7	18.3	30.2	12.6	11.6	9.6	7.3	11.8	26.4	15.8	12.9	17.4±0.21
	SLWI	**12.0**	**11.4**	**11.2**	**27.0**	**21.7**	**14.8**	**21.7**	**11.0**	**9.9**	**8.5**	**6.3**	**10.6**	**16.7**	**11.6**	**9.3**	**13.6±0.17**
0.1	TENT	44.0	43.5	43.8	70.8	78.3	59.9	68.8	42.4	52.0	56.5	30.2	64.7	68.7	63.2	44.7	55.4±1.58
	LAME	45.7	43.7	45.2	72.4	88.0	60.2	87.5	89.1	95.0	99.7	27.1	95.7	95.0	63.2	44.2	70.1±0.04
	RoTTA	43.5	41.2	40.9	68.4	71.1	56.4	64.5	39.1	38.3	38.5	28.3	65.5	67.5	67.2	49.1	52.0±0.01
	SAR	43.9	41.7	40.9	68.7	71.7	54.8	63.3	39.3	39.4	38.8	25.3	44.7	58.1	49.8	39.2	48.0±0.04
	EATA	44.1	42.6	42.6	64.7	68.8	52.1	59.9	37.7	38.0	32.6	23.8	39.6	58.1	51.7	38.4	46.3±0.04
	ROID	40.8	39.3	39.4	55.8	56.1	46.9	54.3	35.8	35.1	30.1	23.6	35.7	49.1	42.2	35.1	41.3±0.03
	SLWI	**39.8**	**37.9**	**37.8**	**51.6**	**48.7**	**41.7**	**45.5**	**33.6**	**32.4**	**28.2**	**22.0**	**34.2**	**40.9**	**37.1**	**31.2**	**37.5±0.02**

Table 4. Average error rates (%) and standard deviations in **the label shifts scenario on D109**. Red fonts indicate performance degradation with respect to Source.

γ	Method	Adaptation Order (→)					Avg.
		clipart	infograph	painting	real	sketch	
–	Source	48.7	72.9	41.2	20.5	56.7	48.0
0.0	TENT	49.2	77.1	51.5	32.6	80.9	58.2±0.04
	LAME	26.0	68.8	**19.2**	**8.0**	**26.7**	29.7±0.15
	RoTTA	48.7	72.8	41.1	20.5	56.6	48.0±0.02
	SAR	48.8	74.5	44.2	20.5	57.0	49.0±0.03
	EATA	48.1	71.8	40.2	21.4	56.5	47.6±0.87
	ROID	25.4	56.1	21.2	10.4	33.3	29.3±0.03
	SLWI	**24.0**	**51.3**	20.4	10.1	30.6	**27.3±0.07**
0.1	TENT	49.1	77.4	51.3	31.7	79.7	57.8±0.06
	LAME	69.8	94.5	57.6	32.7	68.3	64.6±0.25
	RoTTA	48.7	72.7	40.9	20.2	55.3	47.6±0.01
	SAR	48.6	74.0	42.7	20.5	56.8	48.5±0.04
	EATA	48.2	71.6	39.5	19.6	55.0	46.8±0.06
	ROID	35.5	63.7	28.0	12.8	41.7	36.3±0.07
	SLWI	**33.2**	**59.2**	**26.5**	**12.2**	**36.6**	**33.5±0.09**

error rate across all domains on ImageNet-C following the completion of the adaptation process. Notably, the average error rate on the ImageNet test dataset (i.e., the source domain) was minimal, with less than a 1% difference between the methods, indicating the minimal impact of adaptation on the source domain performance. Table 2 presents the performance of all methods on D109. In this case, SLWI also exhibits a superior performance compared to the existing methods, with a significant performance improvement of 2.4% over ROID.

Tables 3 and 4 present the performances of the OTTA methods based on the intensity of label shifts γ. SLWI exhibits consistent performance improvements over the existing OTTA methods across most corruption and natural domains. In particular, for ImageNet-C, SLWI exhibits a remarkable improvement of 3.8% in scenarios with both the highest (i.e., $\gamma = 0.0$) and lower label shift intensities (i.e., $\gamma = 0.1$) compared with ROID. These findings underscore the effectiveness of SLWI in adapting to various challenging environments and maintaining a robust performance across different degrees of label shifts and corruption.

Table 5. Average error rates (%) and standard deviations under various scenarios on ImageNet-C. CS and LS denote covariate and label shifts, respectively.

Model		CS		LS ($\gamma = 0.0$)		LS ($\gamma = 0.1$)	
	Source	ROID	SLWI	ROID	SLWI	ROID	SLWI
ViT	60.2	45.0±0.08	**44.5±0.08**	16.3±0.06	**15.8±0.02**	41.3±0.05	**40.5±0.03**
Swin	64.0	47.2±0.15	**46.3±0.17**	18.1±0.03	**16.6±0.13**	42.1±0.04	**39.4±0.07**
D2V	51.8	44.8±0.04	**42.5±0.03**	17.4±0.21	**13.6±0.17**	41.3±0.03	**37.5±0.02**

4.3. Comparison with Various Models

Table 5 presents the performance of SLWI compared to that of ROID across various models. The results show that SLWI outperformed over ROID for all models tested. These results indicate the applicability of SLWI across a diverse range of models, showing its versatility and effectiveness in enhancing the performance regardless of the underlying model architecture. This adaptability reinforces SLWI's potential as a robust solution that is capable of addressing the diverse challenges encountered in different model structures.

5. Ablation study

5.1. Integration Efficiency with Various OTTA Methods

The proposed framework infers latent weight through a simple operation without backpropagation of the weights obtained from the OTTA procedure. In addition, because the SGD optimizer already calculates the gradient g_t to produce θ_t within the procedure, no additional computation is required in our framework. Owing to these technical advantages, the proposed SLWI framework can be integrated with various TTA methods to offer high computational efficiency. Table 6 lists the average error rates before and after applying SLWI. The results reveal that SLWI, when combined with TENT, EATA, or ROID, demonstrates a significant performance improvement and enhanced stability.

Table 6. Average error rates (%) and standard deviations w/o and w/ SLWI on ImageNet-C. CS and LS denote covariate and label shifts, respectively.

	CS	LS ($\gamma = 0.0$)	LS ($\gamma = 0.1$)
TENT	53.3±0.22	56.2±0.98	55.4±1.58
TENT w/ SLWI	**51.1±0.07**	**27.0±0.07**	**50.8±0.15**
EATA	51.8±0.08	48.2±0.34	46.8±0.06
EATA w/ SLWI	**47.4±0.09**	**20.2±0.12**	**43.9±0.01**
ROID	44.8±0.04	17.4±0.21	41.3±0.03
ROID w/ SLWI	**42.5±0.03**	**13.6±0.17**	**37.5±0.02**

Table 7. Efficiency comparison in covariate shifts scenario on ImageNet-C.

	TENT	TENT w/SLWI	EATA	EATA w/SLWI	ROID	ROID w/SLWI
Avg. GPU time (sec)	159.8	167.7	175.3	178.9	257.5	263.8
Relative time (%)	–	5.0	–	2.0	–	2.4

The performance enhancement capability of SLWI does not require a substantial increase in computational cost. Table 7 lists the average execution times before and after applying SLWI. The best-performing method, ROID, had an average execution time of 257.5s, whereas that of SLWI was 263.8s. These results indicate that only a 2.4% increase in computational effort is required for a considerable performance improvement. Furthermore, when applied to relatively faster methods, such as TENT and EATA, SLWI only requires an additional 5.0% and 2.0% in computational time, respectively. This makes SLWI an efficient and valuable addition to existing OTTA methods, offering a significant performance enhancement with minimal computational overhead.

5.2. Effectiveness of Backtracking

To investigate the effectiveness of SLWI backtracking, we conducted a study in the label shift scenario with $\gamma = 0.0$ on ImageNet-C. We adjusted the backtracking hyperparameters ζ and α and measured their performance, as shown in Figure 4. The results indicate that when backtracking is disabled (i.e., $\zeta = \infty$ and $\alpha = 1.0$), there is notable instability in the performance variability and averages across different seeds.

The left side in Figure 4 shows a substantial enhancement in performance when the stationary level is set below the 95% confidence interval, corresponding to $\zeta = 2$. This result suggests that the proposed framework can effectively detect and address nonlinear weights. The right side in Figure 4 shows that maintaining α within 1.2 to 1.8 facilitated stable performance levels and, conversely, exceeding this range by setting α above 1.8 introduced overfitting due to excessive reliance on past observation, destabilizing the learning process.

The insights garnered from this investigation affirm that nonlinear weights derived from the OTTA procedure can potentially degrade performance. However, through the strategic application of the backtracking mechanism, it is possible to mitigate such risks effectively. The right side in Figure 4 shows that SLWI maintained a stable performance when the degree of backtracking was set between 1.2 and 1.8. These results imply that nonlinear weights derived from the OTTA procedure pose a performance degradation risk and that SLWI effectively mitigates this risk through backtracking, thus playing a vital role in the observed performance enhancement.

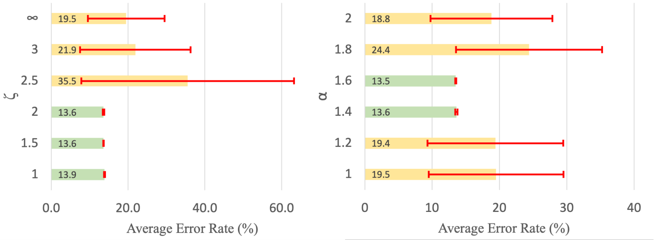


Figure 4. Average error rates and standard deviations (red line) for various ζ and α. $\zeta = \infty$ or $\alpha = 1.0$ indicate that backtracking is not applied.

6. Conclusion

In this study, we addressed the limitations of OTTA methods, which simultaneously perform validation and domain adaptation during the test phase, but fail to comprehensively consider target domain information. We analyzed the standard procedures adopted by existing OTTA methods and proposed the SLWI framework that introduces latent weight capable of continuously accumulating target domain information in the weight space. SLWI leverages the Bayesian filtering framework to infer latent weight, thus mitigating the inherent errors of unreliable weights derived from the unsupervised learning approach of OTTA. We redesigned the framework to accumulate the target and source information and considered the local nonlinearity of unreliable weights, providing a local stationary process. Consequently, SLWI combined with existing OTTA methods achieved outperformed state-of-the-art methods with minimal additional computational costs. These results will contribute to the theoretical research and design of Bayesian filtering for more effectively utilizing information about the target domain in unsupervised domain adaptation.

7. Limitation

One limitation of the proposed framework is the need for hyperparameters to backtrack nonlinear weights. Nevertheless, SLWI exhibited consistent performance improvements across various environments with different distribution shifts and diverse datasets, even with fixed hyperparameters. In future work, we plan to focus on implicitly identifying these hyperparameters to further enhance the adaptability and effectiveness of our approach to dynamically changing domains.

Acknowledgements

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. RS-2023-00302424). This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No. 2020-0-01373, Artificial Intelligence Graduate School Program (Hanyang University)). ※ MSIT : Ministry of Science and ICT

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Abuduweili, A. and Liu, C. Robust online model adaptation by extended kalman filter with exponential moving average and dynamic multi-epoch strategy. *Learning for Dynamics and Control*, pp. 65–74, 2020.

Ansari, A. F., Heng, A., Lim, A., and Soh, H. Neural continuous-discrete state space models for irregularly-sampled time series. *arXiv preprint arXiv:2301.11308*, 2023.

Baevski, A., Hsu, W.-N., Xu, Q., Babu, A., Gu, J., and Auli, M. data2vec: A general framework for self-supervised learning in speech, vision and language. *International Conference on Machine Learning*, pp. 1298–1312, 2022.

Bell, B. M. and Cathey, F. W. The iterated kalman filter update as a gauss-newton method. *IEEE Transactions on Automatic Control*, 38(2):294–297, 1993.

Boudiaf, M., Mueller, R., Ben Ayed, I., and Bertinetto, L. Parameter-free online test-time adaptation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8344–8353, 2022.

Chen, D., Wang, D., Darrell, T., and Ebrahimi, S. Contrastive test-time adaptation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 295–305, 2022.

Cheng, Y., Zhao, W., Liu, C., and Tomizuka, M. Human motion prediction using semi-adaptable neural networks. *American Control Conference*, pp. 4884–4890, 2019.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

Döbler, M., Marsden, R. A., and Yang, B. Robust mean teacher for continual and gradual test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7704–7714, 2023.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Foret, P., Kleiner, A., Mobahi, H., and Neyshabur, B. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020.

Gong, T., Jeong, J., Kim, T., Kim, Y., et al. Note: Robust continual test-time adaptation against temporal correlation. *Neural Information Processing Systems*, 2022.

Gong, T., Kim, Y., Lee, T., Chottananurak, S., and Lee, S.-J. Sotta: Robust test-time adaptation on noisy data streams. *arXiv preprint arXiv:2310.10074*, 2023.

Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. *Deep learning*, volume 1. MIT Press, 2016.

Graves, A., rahman Mohamed, A., and Hinton, G. E. Speech recognition with deep recurrent neural networks. *IEEE international conference on acoustics, speech and signal processing*, pp. 6645–6649, 2013.

Guo, H., Jin, J., and Liu, B. Stochastic weight averaging revisited. *Applied Sciences*, 13(5):2935, 2023.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. *International Conference on Learning Representations*, 2019a.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019b.

Hendrycks, D., Basart, S., Mu, N., Kadavath, S., et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. *International Conference on Computer Vision*, 2021.

Hinton, G. E. et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29: 82–97, 2012.

Huang, H., Gu, X., Wang, H., Xiao, C., Liu, H., and Wang, Y. Extrapolative continuous-time bayesian neural network for fast training-free test-time adaptation. *Advances in Neural Information Processing Systems*, 35:36000–36013, 2022.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *International Conference on Machine Learning*, pp. 448–456, 2015.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., et al. Wilds: A benchmark of in-the-wild distribution shifts. *International Conference on Machine Learning*, pp. 5637–5664, 2021.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25, 2012.

Kurle, R., Cseke, B., Klushyn, A., van der Smagt, P., and Günnemann, S. Continual learning with bayesian neural networks for non-stationary data. In *International Conference on Learning Representations*, 2020. URL https://api.semanticscholar.org/CorpusID:211091599.

Li, A., Boyd, A., Smyth, P., and Mandt, S. Detecting and adapting to irregular distribution shifts in bayesian online learning. In *Neural Information Processing Systems*, 2020. URL https://api.semanticscholar.org/CorpusID:239998047.

Li, Y., Wang, N., Shi, J., Hou, X., and Liu, J. Adaptive batch normalization for practical domain adaptation. *Pattern Recognition*, 80:109–117, 2018.

Li, Z., Malladi, S., and Arora, S. On the validity of modeling sgd with stochastic differential equations (sdes). *Advances in Neural Information Processing Systems*, 34: 12712–12725, 2021.

Liang, J., Hu, D., and Feng, J. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning*, pp. 6028–6039. PMLR, 2020.

Liu, Z., Lin, Y., Cao, Y., Hu, H., et al. Swin transformer: Hierarchical vision transformer using shifted windows. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10012–10022, 2021.

Mancini, M., Karaoguz, H., Ricci, E., Jensfelt, P., et al. Kitting in the wild through online domain adaptation. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1103–1109, 2018.

Marsden, R. A. and Döbler, M. test-time-adaptation. https://github.com/mariodoebler/test-time-adaptation, 2022.

Marsden, R. A., Döbler, M., and Yang, B. Universal test-time adaptation through weight ensembling, diversity weighting, and prior correction. *arXiv preprint arXiv:2306.00650*, 2023.

Murphy, K. P. *Probabilistic machine learning: Advanced topics*. MIT press, 2023.

Niu, S., Wu, J., Zhang, Y., Chen, Y., et al. Efficient test-time model adaptation without forgetting. *International Conference on Machine Learning*, pp. 16888–16905, 2022.

Niu, S., Wu, J., Zhang, Y., Wen, Z., et al. Towards stable test-time adaptation in dynamic wild world. *ArXiv*, abs/2302.12400, 2023.

Peng, X., Bai, Q., Xia, X., Huang, Z., et al. Moment matching for multi-source domain adaptation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1406–1415, 2019.

Press, O., Schneider, S., Kümmerer, M., and Bethge, M. Rdumb: A simple approach that questions our progress in continual test-time adaptation. *Advances in Neural Information Processing Systems*, 36, 2024.

Puskorius, G. V. and Feldkamp, L. A. Parameter-based kalman filter training: Theory and implementation. *Kalman filtering and neural networks*, pp. 23–67, 2001.

Quinonero-Candela, J., Sugiyama, M., Schwaighofer, A., and Lawrence, N. D. *Dataset shift in machine learning*. MIT Press, 2008.

Ruder, S. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

Särkkä, S. and Solin, A. *Applied stochastic differential equations*, volume 10. Cambridge University Press, 2019.

Särkkä, S. and Svensson, L. Bayesian filtering and smoothing. 2023. URL https://api.semanticscholar.org/CorpusID:261435737.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Steiner, A., Kolesnikov, A., Zhai, X., Wightman, R., et al. How to train your vit? data, augmentation, and regularization in vision transformers. *arXiv preprint arXiv:2106.10270*, 2021.

Sun, S., Zhang, B., Xie, L., and Zhang, Y. An unsupervised deep domain adaptation approach for robust speech recognition. *Neurocomputing*, 257:79–87, 2017.

Wang, D., Shelhamer, E., Liu, S., Olshausen, B., and Darrell, T. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

Wang, H., Ge, S., Lipton, Z., and Xing, E. P. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pp. 10506–10518, 2019.

Wang, Q., Fink, O., Van Gool, L., and Dai, D. Continual test-time domain adaptation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7201–7211, 2022.

Yuan, L., Xie, B., and Li, S. Robust test-time adaptation in dynamic scenarios. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15922–15932, 2023.

Zhang, M., Levine, S., and Finn, C. Memo: Test time robustness via adaptation and augmentation. *Advances in Neural Information Processing Systems*, 35:38629–38642, 2022.

Zhou, A. and Levine, S. Bayesian adaptation for covariate shift. *Advances in Neural Information Processing Systems*, 34:914–927, 2021.

Zhou, Z., Guo, L.-Z., Jia, L.-H., Zhang, D., and Li, Y.-F. Ods: test-time adaptation in the presence of open-world data shift. *International Conference on Machine Learning*, pp. 42574–42588, 2023.

A. Framework Details and Derivations

To ensure the completeness of the paper, we detail the background and derivation process of the proposed framework design in this section. Algorithm 3 provides utility functions for the entire framework.

Algorithm 3 Stationary Latent Weight Inference

```
function Prediction($\mu_{t-1|t-1}, \phi_0, \sigma^2_{t-1|t-1}; a, q$)
    $\mu_{t|t-1} \leftarrow a\mu_{t-1|t-1} + (1-a)\phi_0$
    $\sigma^2_{t|t-1} \leftarrow a^2\sigma^2_{t-1|t-1} + q$
    Return $\mu_{t|t-1}, \sigma^2_{t|t-1}$
end function

function Backtracking($\theta_t, g_t, \mu_{t|t-1}, \sigma^2_{t-1|t-1}, \log p_{g,t-1}(\theta_{0:t-1}), \Delta_{0:t-1}; a, \zeta$)
    $c_t \leftarrow 1$
    $\log p_{g,t-1}(\theta_t|\theta_{0:t-1}) \leftarrow \mathcal{N}(\theta_t|\mu_{t|t-1}, a^2\sigma^2_{t-1|t-1} + 1)$
    $\Delta_t \leftarrow \log p_{g,t-1}(\theta_t|\theta_{0:t-1}) - \frac{1}{t}\log p_{g,t-1}(\theta_{0:t-1})$
    if $|S(t, \Delta_t, \Delta^2_{0:t-1})| \geq \zeta$ then
        $c_t \leftarrow 1/\alpha$
    end if
    $\hat{\theta}_t \leftarrow \theta_t - c_t g_t$
    $\log p_{g,t-1}(\theta_{0:t}) \leftarrow \log p_{g,t-1}(\theta_{0:t-1}) + \log p_{g,t-1}(\theta_t|\theta_{0:t-1})$
    $\Delta^2_{0:t} \leftarrow \Delta^2_{0:t-1} + \Delta^2_t$
    Return $\hat{\theta}_t, \log p_{g,t-1}(\theta_{0:t}), \Delta^2_{0:t}$
end function

function Update($\hat{\theta}_t, \mu_{t|t-1}, \sigma^2_{t-1|t-1}, \sigma^2_{t|t-1}; a$)
    $\kappa_t = \sigma^2_{t|t-1}(a^2\sigma^2_{t-1|t-1} + 1)^{-1}$
    $\mu_{t|t} \leftarrow \mu_{t|t-1} + \kappa_t\,(\hat{\theta}_t - \mu_{t|t-1})$
    $\sigma^2_{t|t} \leftarrow \sigma^2_{t|t-1} - \kappa_t\sigma^2_{t|t-1}$
    Return $\mu_{t|t}, \sigma^2_{t|t}$
end function

function Filter($t, \theta_t, g_t, \phi_0, \mu_{t-1|t-1}, \sigma^2_{t-1|t-1}, \log p_{g,t-1}(\theta_{0:t-1}), \Delta_{0:t-1}; \Omega, \omega$)
    $a, \_, q \leftarrow \Omega$
    $\zeta, \alpha \leftarrow \omega$
    $\mu_{t|t-1}, \sigma^2_{t|t-1} \leftarrow$ Prediction($\mu_{t-1|t-1}, \phi_0, \sigma^2_{t-1|t-1}; a, q$)
    if $t > 1$ then
        $\log p_{g,t-1}(\theta_{0:t}), \Delta^2_{0:t}, \hat{\theta}_t \leftarrow$ Backtracking($\theta_t, g_t, \mu_{t|t-1}, \sigma^2_{t-1|t-1}, \log p_{g,t-1}(\theta_{0:t-1}), \Delta_{0:t-1}; a, \zeta$)
    else
        $\log p_{g,t-1}(\theta_{0:t}), \Delta^2_{0:t}, \hat{\theta}_t \leftarrow \log p_{g,t-1}(\theta_{0:0}), \Delta^2_{0:0}, \theta_t$
    end if
    $\mu_{t|t}, \sigma^2_{t|t} \leftarrow$ Update($\hat{\theta}_t, \mu_{t|t-1}, \sigma^2_{t-1|t-1}, \sigma^2_{t|t-1}; a$)
    Return $\mu_{t|t}, \mu_{t|t-1}, \sigma^2_{t|t}, \log p_{g,t-1}(\theta_{0:t}), \Delta^2_{0:t}$
end function
```

A.1. Stationary Linear Gaussian Model

The OTTA method, operating in the target domain, often experiences high uncertainty due to the significant performance drop of the source model and the adoption of an unsupervised learning approach without true labels. This uncertainty generally leads OTTA's SGD-based process to adopt the learning rate between 1.0^{-4} and 1.0^{-5}, which are about 100 times smaller than the learning rate between 1.0^{-2} and 1.0^{-3} used for training the source model (Wang et al., 2020; 2022; Yuan et al., 2023; Niu et al., 2022; Steiner et al., 2021). Furthermore, as we use a method that learns only a tiny fraction of the weights in the model (Niu et al., 2023; Marsden et al., 2023), the learning rate is 0 for most of the weights. When the learning rate is very small, the SGD process can be approximated as a stochastic differential equation (SDE) (Li et al., 2021). The marginal density of this SDE is governed by Fokker-Planck-Kolmogorov (FPK) equation (Särkkä & Solin, 2019). Solutions to this equation frequently involve the Gaussian assumed density approximation (Särkkä & Svensson, 2023). The posterior distribution of Bayesian filters using linear Gaussian models can be a target of the Gaussian assumed density and satisfy the FPK equation (Ansari et al., 2023). Based on this theoretical foundation, we adopt linear Gaussian models to model the weight evolution in the OTTA procedure. Considering the minute weight changes over time due to the small learning rate, we assume the weight changes closely resemble a local stationary process. Thus, we design our linear Gaussian models for the transition and emission as stationary linear systems, as indicated in Eqs. (5) and (6). In addition, such stationary settings help to stabilize the process (Murphy, 2023).

A.2. Inference using Bayesian Filtering Equations

The process of inferring the latent-weight posterior using transition and emission models in Bayesian filtering is a recursive method. Given the one-time step previous posterior distribution derived from past domains, denoted as $p(\phi_{t-1}|\theta_{1:t-1}, g_{1:t-1}, \phi_0)$, this serves as the prior distribution for the next step. Along with Eq. (5), we construct the joint distribution and then perform marginalization to obtain the one-step-ahead predictive distribution. According to Lemma A.2 and A.3 from (Särkkä & Svensson, 2023), this distribution is formulated as follows:

$$\begin{aligned} p(\phi_t|\theta_{1:t-1}, g_{1:t-1}, \phi_0) &= \int p(\phi_t|\phi_{t-1}, \phi_0) p(\phi_{t-1}|\theta_{1:t-1}, g_{1:t-1}, \phi_0) d\phi_{t-1} \\ &= \mathcal{N}(\phi_t|\mu_{t|t-1}, \sigma^2_{t|t-1}), \end{aligned} \tag{16}$$

where $\mu_{t|t-1} = a\mu_{t-1|t-1} + (1-a)\phi_0, \sigma^2_{t|t-1} = a^2\sigma^2_{t-1|t-1} + q$. According to Lemma A.2 and A.3 from Särkkä & Svensson 2023, the posterior distribution for the latent weight using the predictive distribution and Eq. (6) is

$$p(\phi_t|\theta_{1:t}, g_{1:t}, \phi_0) = \frac{p(\theta_t|\phi_t, g_t) p(\phi_t|\theta_{1:t-1}, g_{1:t-1}, \phi_0)}{\int p(\theta_t|\phi_t, g_t) p(\phi_t|\theta_{1:t-1}, g_{1:t-1}, \phi_0)} = \mathcal{N}(\phi_t|\mu_{t|t}, \sigma^2_{t|t}), \tag{17}$$

where $\kappa_t = \sigma^2_{t|t-1}(a^2\sigma^2_{t-1|t-1} + 1)^{-1}, \mu_{t|t} = \mu_{t|t-1} + \kappa_t(\theta_t - c_t g_t - \mu_{t|t-1})$ and $\sigma^2_{t|t} = \sigma^2_{t|t-1} - \kappa_t \sigma^2_{t|t-1}$. According to Lemma A.2 and A.3 from Särkkä & Svensson 2023, the denominator term of the posterior predictive distribution is marginalized as follows:

$$\begin{aligned} p_{c_t}(\hat{\theta}_t|\theta_{0:t-1}, g_{1:t-1}) = p(\theta_t|\theta_{0:t-1}, g_{1:t}) &= \int p(\theta_t|\phi_t, g_t) p(\phi_t|\theta_{0:t-1}, g_{1:t-1}) d\phi_t \\ &= \mathcal{N}(\theta_t|\mu_{t|t-1} - (1-c_t)g_t, a^2\sigma^2_{t-1|t-1} + 1), \end{aligned} \tag{18}$$

where $\hat{\theta}_t = \theta_{t-1} - c_t g_t$. For the initial condition $c_t = 1$, $p(\theta_t|\phi_t, g_t) = p(\theta_t|\phi_t)$, and then $p_{c_t}(\hat{\theta}_t|\theta_{0:t-1}, g_{1:t-1})$ becomes $p_{g,t-1}(\theta_t|\theta_{0:t-1})$. Thus, we can naturally calculate Eq. (11) when determining the posterior distribution for the latent weight. This convenience leads to the technical advantage of requiring minimal additional computation for backtracking as described in Section 3.2.

A.3. Log-Posterior Predictive Normalization for Non-Stationarity Detection

Nonlinearity in weight changes can be defined as outliers in the joint distribution of weights. The log-posterior predictive distribution encapsulates the difference between distributions of the entire weights, including the current observation and the previous time step:

$$\log p_{g,t-1}(\theta_t|\theta_{0:t-1}) = \log p_{g,t-1}(\theta_t, \theta_{0:t-1}) - \log p_{g,t-1}(\theta_{0:t-1}), \tag{19}$$

where $\log p_{g,t-1}(\theta_{0:t-1}) = \log p(\theta_1|\theta_0) + \sum_{\tau=2}^{t-1} \log p_{g,t-1}(\theta_\tau|\theta_{0:\tau-1})$. The first term represents the joint distribution of all weights, including the current observation, while the second term signifies the joint distribution of weights up to one-time step prior. A large difference indicates that the current weights are outliers, requiring a reduction in θ_t to approximate θ_{t-1}, necessitating c_t to be less than 1 in $\hat{\theta}_t$. However, since the log-likelihood is not normalized over time, comparing across different times is challenging, making it difficult to set a threshold for detecting outliers. To address this, we adopt a batch normalization-like approach (Ioffe & Szegedy, 2015), computing running statistics and normalizing the log-likelihood for

$t > 1$ as follows:

$$\begin{aligned} S(t, \Delta_t, \Delta^2_{0:t-1}) &= \frac{\Delta_t}{\sqrt{\frac{1}{t-1}\Delta^2_{0:t-1}}} \\ &= \frac{\log p_{g,t-1}(\theta_t|\theta_{0:t-1}) - \frac{1}{t}\log p_{g,t-1}(\theta_{0:t-1})}{\sqrt{\frac{1}{t-1}\sum_{\tau=1}^{t-1}(\log p_{g,t-1}(\theta_t|\theta_{0:\tau-1}) - \frac{1}{\tau}\log p_{g,t-1}(\theta_{0:\tau-1}))^2}}, \end{aligned} \tag{20}$$

where $\Delta_t = \log p_{g,t-1}(\theta_t|\theta_{0:t-1}) - \frac{1}{t}\log p_{g,t-1}(\theta_{0:t-1})$ and $\Delta^2_{0:t-1} = \sum_{\tau=1}^{t-1}\Delta^2_\tau$ with initial conditions $\log p_{g,t-1}(\theta_{0:0}) = 0, \Delta^2_{0:0} = 0$. From a batch normalization perspective, $\frac{1}{t}\log p_{g,t-1}(\theta_{0:t-1})$ is the running mean, and $\frac{1}{t-1}\Delta^2_{0:t-1}$ is the running variance. With this score and an arbitrary threshold $\zeta \geq 0$, we can perform decision thresholding as follows:

$$\left|\frac{\Delta_t}{\sqrt{\frac{1}{t-1}\Delta^2_{0:t-1}}}\right| < \zeta, \tag{21}$$

$$\frac{1}{t}\log p_{g,t-1}(\theta_{0:t-1}) - \zeta\sqrt{\frac{1}{t-1}\Delta^2_{0:t-1}} < \log p_{g,t-1}(\theta_t|\theta_{0:t-1}) < \frac{1}{t}\log p_{g,t-1}(\theta_{0:t-1}) + \zeta\sqrt{\frac{1}{t-1}\Delta^2_{0:t-1}}, \tag{22}$$

where ζ serves a role similar to a confidence level in a normal distribution. We set weights $\hat{\theta}_t$ that do not meet this criterion as outliers and perform backtracking with c_t less than 1 as described in Section 3.3 and Eq. (14).

A.4. Transfer Model for Plug-in Approximation

Latent weight information is transferred to the OTTA procedure via the transfer model $p(\theta_t|\phi_t) = \mathcal{N}(\theta_t|\phi_t, r_t)$. Owing to the plug-in approximation, the latent-weight posterior variance obtained from $p(\phi_t|\theta_{1:t-1}, g_{1:t-1}, \phi_0)$ and the transfer model must be sufficiently small. According to Lemma A.2 and A.3 from (Särkkä & Svensson, 2023), the mean and variance of the posterior are calculated as follows:

$$\mu_t = \mu_{t|t-1} + \kappa'_t(\hat{\theta}_t - \mu_{t|t-1}), \tag{23}$$

$$\sigma^2_t = (1 - \kappa'_t)\sigma^2_{t|t-1}, \tag{24}$$

where the Kalman gain is $\kappa'_t = \sigma^2_{t|t-1}(\sigma^2_{t|t-1} + r_t)^{-1}$. By setting the proportionality constant $s \in \mathbb{R}$ such that r_t is proportional to $\sigma^2_{t|t-1}$, the Kalman gain becomes time-independent as $s(s+1)^{-1}$. Finally, the point estimated weight is calculated as:

$$\hat{\phi}_t = \mu_{t|t-1} + b(\hat{\theta}_t - \mu_{t|t-1}), \tag{25}$$

where $b = s(s+1)^{-1}$. This result was used in Section 3.4.

B. Additional Experiments Details

Our experiments were conducted using a single NVIDIA GeForce RTX 3090 GPU, and we provide specific experimental settings in this section. The evaluation metrics involved averaging and calculating the standard deviation of error rates for random seeds 0-4.

Datasets We focused on the diverse classes, corruption, and natural distribution shifts prevalent in the wild-world (Niu et al., 2023). ImageNet-C, a standard TTA benchmark, evaluates robustness against corruption. ImageNet consists of $1,281,167$ training and $50,000$ testing data. ImageNet-C applies 15 types of corruption at five severity levels to ImageNet, with each corruption considered a domain. We selected severity level 5 for our experiments. D109 deals with natural distribution shifts, comprising five domains (clipart, infograph, painting, real, and sketch) based on DomainNet, including 109 overlapping classes with ImageNet. We also used Rendition (Hendrycks et al., 2021) and Sketch (Wang et al., 2019) datasets for other natural shifts. Rendition includes $30,000$ images from various artistic renderings of 200 ImageNet classes, collected from Flickr and filtered by Amazon MTurk annotators. The Sketch dataset contains $50,000$ images, with 50 images for each of the $1,000$ ImageNet classes, sourced from Google image queries with the term "sketch of" in a "black and white" color scheme.

Compared Methods We compared SLWI with various contemporary OTTA methods. TENT updates trainable weights specified by entropy minimization loss. LAME modifies model outputs, not weights, during testing for adaptation to label distribution shifts. RoTTA uses a student-teacher approach with a cross-entropy objective function and data augmentation. EATA employs an entropy-based objective function, excluding samples with high entropy based on an entropy constant threshold. Similarly, SAR adapts models using sample exclusion methods, and SAM (Foret et al., 2020) to prevent settling into sharp local optima. ROID uses an entropy objective function that excludes some samples based on a diversity score of label distribution.

All experiments strictly adhered to the hyperparameters of each method as per the existing benchmark (Marsden & Döbler, 2022), which references the official implementations and hyperparameters reported in the original papers. Where hyperparameters for a specific dataset or model were not provided for a method, we adjusted them accordingly. The experiments typically used an SGD optimizer with momentum 0.9, with learning rates set at $0.00001/0.00025$ for D2V/ViT models and the same for Swin as for ViT. For EATA, the learning rate was 0.000005 for D2V models. For the SLWI framework, parameters a and b were set to 0.99 for all models. Another parameter, q, was set to 0.00025, 0.005, and 0.001 for ViT, Swin, and D2V models. The degree of backtracking α was set at 2.5 for ViT and Swin and 1.4 for D2V.

C. Additional Experiments: General Unsupervised Domain Adaptation Performance in One-Epoch Adaptation Scenario

Table 8. Average error rates (%) and standard deviations in **the one-epoch adaptation scenario on ImageNet-C**. Red fonts indicate performance degradation with respect to Source.

Method	NOISE			BLUR				WEATHER				DIGITAL				Avg.
	gaussian	shot	impulse	defocus	glass	motion	zoom	snow	frost	fog	bright	contrast	elastic	pixelate	jpeg	
Source	43.9	43.3	43.4	69.7	78.3	59.6	69.1	40.1	44.3	36.3	26.5	50.6	67.6	60.6	43.4	51.8
TENT	44.8	43.9	43.9	70.3	77.3	58.5	67.6	40.8	41.4	37.8	26.1	47.2	66.7	60.2	43.6	51.3±0.09
LAME	44.6	43.9	44.1	70.1	78.6	60.0	69.6	40.8	45.4	38.4	27.4	52.3	68.3	61.2	44.2	52.6±0.14
RoTTA	43.5	42.9	42.7	69.8	77.8	59.4	68.7	39.7	42.8	36.0	26.3	49.9	67.1	60.4	43.2	51.3±0.03
SAR	44.8	44.2	44.0	70.9	78.4	59.0	68.3	40.7	41.0	37.9	26.1	48.7	66.6	60.1	43.2	51.6±0.09
EATA	44.5	43.7	43.5	71.1	75.6	59.5	69.0	41.7	44.1	40.8	26.5	50.2	66.2	61.3	44.2	52.1±0.10
ROID	42.9	42.3	42.0	64.7	70.4	54.3	62.3	38.4	37.2	34.2	24.5	42.6	59.7	55.1	40.3	47.4±0.05
SLWI	**41.9**	**41.0**	**40.9**	**60.4**	**64.6**	**50.3**	**59.2**	**36.8**	**36.0**	**33.5**	**23.5**	**41.4**	**55.0**	**50.6**	**38.4**	**44.9±0.03**

Table 9. Average error rates (%) and standard deviations in **the one-epoch adaptation scenario on D109**. Red fonts indicate performance degradation with respect to Source.

Method	clipart	infograph	painting	real	sketch	Avg.
Source	48.7	72.9	41.2	20.5	56.7	48.0
TENT	55.3	79.1	47.0	23.5	63.1	53.6±0.17
LAME	98.0	99.1	97.7	97.2	98.4	98.1±0.07
RoTTA	46.8	71.9	40.3	20.1	55.1	46.8±0.03
SAR	48.9	73.0	41.0	20.6	57.1	48.1±0.02
EATA	47.4	71.7	39.8	19.8	56.0	47.0±0.02
ROID	44.6	70.0	38.0	19.3	53.2	45.0±0.01
SLWI	**41.3**	**67.3**	**36.3**	**18.7**	**49.3**	**42.6±0.06**

Table 10. Average error rates (%) and standard deviations in **the one-epoch adaptation scenario on Rendition and Sketch**. Red fonts indicate performance degradation with respect to Source.

Method	Rendition	Sketch
	Avg.	Avg.
Source	46.6	60.4
TENT	46.5±0.03	60.6±0.02
LAME	86.4±0.35	86.7±0.45
RoTTA	46.5±0.02	60.1±0.03
SAR	46.2±0.05	60.5±0.05
EATA	46.2±0.04	59.6±0.05
ROID	41.8±0.12	56.2±0.04
SLWI	**39.0±0.06**	**53.0±0.07**

To assess general unsupervised domain adaptation performance, we considered a one-epoch adaptation scenario, where input and label data are not time-correlated, meaning domains are randomly mixed and fed to the model. Each sample was fed to the model only once, making it a typical domain adaptation scenario with only one training epoch. We compared OTTA methods in a one-epoch adaptation scenario. Table 8 shows the performance of each method on ImageNet-C. SLWI showed a performance improvement of 2.5% over the latest ROID. Tables 9 and 10 display the performance of various methods under different natural shift scenarios. Compared to ROID, SLWI demonstrated performance improvements of 2.4%, 2.8%, and 3.2% on the D109, Rendition, and Sketch datasets, respectively. These results empirically indicated that SLWI consistently improved performance on various datasets, even under general unsupervised domain adaptation situations.

D. Additional Ablation Study: Valid Bayesian Filtering Parameters

Table 11. Average error rates (%) and standard deviations in **the covariate scenario on ImageNet-C** along with parameters of SLWI's Bayesian filtering.

Parameter	SLWI					Source
a	0.9999	0.9995	0.99	0.9	0.8	51.8
	42.6±0.13	42.5±0.11	**42.5±0.03**	43.5±0.08	43.6±0.04	
b	0.9999	0.9995	0.99	0.9	0.8	
	42.6±0.14	**42.3±0.11**	42.5±0.03	47.2±0.04	49.2±0.03	
q	0.00025	0.0005	0.001	0.01	0.1	
	43.1±0.07	42.9±0.08	42.5±0.03	42.5±0.19	**42.3±0.15**	

We guide setting Bayesian filtering parameters for SLWI. The transition model aims to recover the latent weight, which may be compromised by unreliable target weights, through the source weight. Since excessive recovery can hinder the utilization of target domain information, values near 1 should be adopted. The parameter b, due to the plug-in approximation, should also be close to 1. We set q to be greater than 0 and less than 0.5, ensuring that the variance of the transition model remains smaller than that of the emission model.

Table 11 lists the performance changes according to variations in each parameter. The results showed that the performance slightly declined when a exceeded 0.99 and more significantly when b exceeded 0.99. This tendency indicated the importance of the constraints imposed by the plug-in approximation on these parameters. Variations in q did not significantly affect the average error rate but impacted stability, with the highest stability observed at $q = 0.001$. Consequently, within the guidelines we provided, SLWI demonstrated robust performance.

E. Related Works: Bayesian Inference for DNNs

Bayesian filtering, mainly based on linear Gaussian models and known as Kalman filtering, is a Bayesian inference method for recursively predicting and updating observations over time (Särkkä & Solin, 2019; Särkkä & Svensson, 2023; Kurle et al., 2020; Li et al., 2020). Kalman filtering is effectively used in natural observation scenarios, such as object tracking (Cheng et al., 2019; Abuduweili & Liu, 2020). To better capture nonlinear changes in non-stationary environments, extensions like the Extended Kalman Filter (EKF) and Iterated EKF have been developed (Bell & Cathey, 1993). These methods have been applied to Kalman filtering of model outputs of nonlinear DNNs (Puskorius & Feldkamp, 2001). The SLWI framework differs in that it performs Bayesian filtering on the weights of DNNs, as opposed to the model outputs addressed in traditional EKF applications. Unlike EKF, where the Kalman gain increases with more significant nonlinearity, SLWI employs backtracking to adjust observations when nonlinearity exceeds a certain level. This approach assumes that excessive nonlinearity in observations indicates errors, necessitating adjustments. Empirically, we demonstrated in Section 5.2 that significant performance degradation occurs when the non-stationary interval is increased, i.e., when a certain level of nonlinearity is accepted. These results suggest that maintaining weights within a stationary range is crucial for stable unsupervised domain adaptation, even when DNNs handle non-stationary streaming data.

On the other hand, the particle filter, another Bayesian inference method addressing time series observation, is used for estimating states in non-Gaussian and nonlinear systems. Recently, it was applied to domain adaptation for streaming data (Huang et al., 2022). This approach requires offline learning using source and target data to train the weights' distribution parameters and their importance using DNNs. During the real-time testing phase, weights are sampled and combined with the learned importance scores to estimate the posterior distribution, effectively performing a weighted ensemble. However, unlike SLWI, this method necessitates prior offline learning using source and target data. The proposed framework is an online test-time adaptation method that does not require access to source data or prior offline learning. Additionally, SLWI, through plug-in approximation and transfer models, avoids the weight sampling process, allowing for much faster computations. It is also memory-efficient, as it does not store multiple weights in memory but only uses additional latent weights.